

What is claimed is:

1. In a computer network having a plurality of nodes, a file replication system

comprising:

means, distributed on each of said nodes, capable of receiving a file in any one of  
said nodes; and,

means, distributed throughout said plurality of nodes and responsive to receiving  
said file in a certain one of said nodes, for replicating said file in all other of said nodes.

2. The system of claim 1 wherein said file is an updated file and wherein said  
replicating means includes:

additional replication means for replicating said updated file in all other of said  
nodes in a manner that is network-topology independent.

3. The system of claim 2 and wherein said replicating means includes:

additional replication means for replicating said updated file in all other of said  
nodes in a manner that avoids a single point of failure.

4. The system of claim 2 and wherein said certain one of said nodes is the originator  
node.

5. The system of claim 4 and wherein said replicating means further comprises:

1 means for establishing another of said nodes as master node, said plurality of  
2 nodes except for both said originator node and said master node being slave nodes;  
3 means for storing said updated file on said master node as a backup file; and,  
4 in each of said slave nodes, means for updating a particular file corresponding to  
5 said updated file.

6  
7 6. The system of claim 5 and further comprising:

8 means, included within said master node, for communicating both creation of said  
9 backup file to said originator node and availability of said backup file to said slave nodes.

10  
11 7. The system of claim 6 and further comprising:

12 said originator node including means, responsive to operation of said  
13 communicating means, for publishing a representation of said updated file to said slave  
14 nodes; and

15 means, within each of said slave nodes and responsive to operation of said  
16 publishing means, for commanding its respective node to obtain said updated file from  
17 said originator node.

18  
19 8. The system of claim 6 and wherein operation of said communicating means  
20 communicating said creation of said backup file to said originator node comprises a  
21 success note.

22  
23 9. The system of claim 7 and further comprising:

1           said originator node including means for establishing a updated file version  
2   variable as said representation of said file; and  
3           said publishing means includes means for publishing said updated file version  
4   variable to said slave nodes.

5  
6   10.    The system of claim 9 and further comprising:

7           said replicating means including means for establishing a particular file version  
8   variable corresponding to said particular file;

9           object observer means, in said each of said slave nodes, for observing change  
10   from said particular file version variable to said updated file version variable; and,

11          said commanding means including means, responsive to operation of said object  
12   observer means, for downloading said updated file from said originator node into said  
13   particular file in said each of said slave nodes.

14  
15   11.    The system of claim 5 and wherein said replicating means comprises:

16          local workspace means for receiving said updated file in said originator node;  
17   and,

18          global workspace means, operatively coupled to said local workspace means, for  
19   receiving said updated file from said local workspace means in preparation to download  
20   said updated file to any of said slave nodes upon request from said any of said slave  
21   nodes.

22  
23   12.    The system of claim 11 and wherein said replicating means further comprises:

1 at least one data file including a data word with its corresponding file version  
2 variable.

3  
4 13. The system of claim 5 and further comprising:

5 means for associating a file version variable with said updated file;

6 said storing means comprising:

7 local workspace means for receiving said updated file in said master  
8 node;

9 means for error checking said file version variable to confirm validity of  
10 said file version variable;

11 global workspace means adapted to receive said updated file from said  
12 local workspace means; and,

13 means, responsive to operation of said error checking means confirming  
14 said validity, for transferring said updated file to said global workspace means;  
15 and,

16 means, responsive to operation of said transferring means, for communicating  
17 both creation of said backup file to said originator node and availability of said backup  
18 file to said slave nodes.

19  
20 14. The system of claim 11 and wherein said local workspace means further includes:

21 multiple source means for receiving said updated file from a new-data-supplier  
22 group consisting of a network user, a security provider, and other providers.

15. The system of claim 14 and wherein said multiple source receiving means includes additional means for receiving additional updated files from said new-data-supplier group.

16. The system of claim 2 and wherein said receiving means is further capable of receiving multiple updated files, each of said files being received from a different network user.

17. The system of claim 5 wherein said replicating means includes means for establishing a particular file version variable corresponding to said particular file, said system further comprising:

polling means for allowing said each of said slave nodes to periodically poll said master node to determine if said particular file contents matches said updated file contents; and,

synchronizing means, responsive to operation of said polling means determining that said particular file contents do not match said updated file contents, for synchronizing said particular file contents with said updated file contents.

18. The system of claim 17 further comprising:

means for establishing a DPS version number to identify the current version of DPS in said network;

means for establishing a DDB version number to identify the current version of DDB in said network;

1 first means for comparing said DPS version number on said each of said slave  
2 nodes with said DPS version number on said master node to obtain a respective DPS  
3 version number match;

4 said synchronizing means,

5 responsive to operation of said first comparing means obtaining said  
6 respective DPS version number match on each of certain of said slave nodes for  
7 terminating further operation of said synchronizing means with respect to the  
8 current said poll on said each of said certain of said slave nodes, and

9 responsive to operation of said first comparing means not obtaining said  
10 respective DPS version number match on each of the remainder of said slave  
11 nodes for achieving said DDB version number match on said each of the  
12 remainder of said slave nodes;

13 second means, responsive to operation of said first comparing means not  
14 obtaining said respective DPS version number match, for comparing said particular file  
15 version variable on said each of the remainder of said slave nodes with said updated file  
16 version variable on said master node to obtain a respective file version variable match;  
17 and,

18 said synchronizing means,

19 responsive to operation of said second comparing means obtaining a file  
20 version variable match on each of a portion of said remainder of said slave nodes  
21 for terminating operation of said synchronizing means with respect to the current  
22 said poll on said each of said portion, and

1 responsive to operation of said second means not obtaining a file version  
2 variable match on each of the remaining portion of said remainder of said slave  
3 nodes for achieving said file version variable match on said each of the remaining  
4 portion;  
5 whereby said particular file contents matches said updated file contents in said  
6 each of said slave nodes.

7  
8 19. The system of claim 5 wherein said replicating means includes means for  
9 establishing a particular file version variable corresponding to said particular file, and  
10 wherein each of said plurality of nodes is a storage system having storage media on  
11 which both said particular file version variable and said particular file contents are stored,  
12 said system further comprising:

13 polling means, for comparing said particular file version variable stored on said  
14 storage media in said each of said nodes with said particular file version variable stored  
15 elsewhere in its respective node to determine a particular file version variable match for  
16 said each of said nodes; and

17 synchronizing means, responsive to operation of said polling means determining  
18 said particular file version variable match was not achieved for certain of said nodes for  
19 achieving said particular file version variable match for each of said certain nodes.

20  
21 20. The system of claim 19 and wherein said storage media is at least one storage  
22 disk.

21. The system of claim 7 further comprising:

means, responsive to operation of said commanding means not obtaining said updated file from said originator node, for further commanding said respective node to obtain said backup file from said master node.

22. The system of claim 13 further comprising:

means, responsive to operation of said error checking means not confirming said validity, for flagging an error, stopping operation of said file replicating means on said updated file, and preparing said file replication system to receive a next successive updated file.

23. A computer program product for use in a computer network having a plurality of nodes, said computer program product including a computer usable medium having computer readable program code thereon for file replication, said program code comprising:

program code, distributed on each of said nodes, capable of receiving a file in any one of said nodes; and,

program code, distributed throughout said plurality of nodes and responsive to receiving said file in a certain one of said nodes, for replicating said file in all other of said nodes.



1 24. The computer program product of claim 23 wherein said file is an updated file  
2 and wherein said replicating program code includes:

3 additional replication program code for replicating said updated file in all other of  
4 said nodes in a manner that is network-topology independent.

5  
6 25. The computer program product of claim 24 and wherein said replicating program  
7 code includes:

8 additional replication program code for replicating said updated file in all other of  
9 said nodes in a manner that avoids a single point of failure.

10  
11 26. The computer program product of claim 24 and wherein said certain one of said  
12 nodes is the originator node.

13  
14 27. The computer program product of claim 26 and wherein said replicating program  
15 code further comprises:

16 program code for establishing another of said nodes as master node, said plurality  
17 of nodes except for both said originator node and said master node being slave nodes;

18 program code for storing said updated file on said master node as a backup file;

19 and,

20 in each of said slave nodes, program code for updating a particular file  
21 corresponding to said updated file.

22  
23 28. The computer program product of claim 27 and further comprising:

1 program code, included within said master node, for communicating both creation  
2 of said backup file to said originator node and availability of said backup file to said slave  
3 nodes.

4  
5 29. The computer program product of claim 28 and further comprising:

6 said originator node including program code, responsive to operation of said  
7 communicating program code, for publishing a representation of said updated file to said  
8 slave nodes; and

9 program code, within each of said slave nodes and responsive to operation of said  
10 publishing program code, for commanding its respective node to obtain said updated file  
11 from said originator node.

12  
13 30. The computer program product of claim 28 and wherein operation of said  
14 communicating program code communicating said creation of said backup file to said  
15 originator node comprises a success note.

16  
17 31. The computer program product of claim 29 and further comprising:

18 said originator node including program code for establishing an updated file  
19 version variable as said representation of said updated file; and

20 said publishing program code includes program code for publishing said updated  
21 file version variable to said slave nodes.

22  
23 32. The computer program product of claim 31 and further comprising:

1           said replicating program code including program code for establishing a particular  
2 file version variable corresponding to said particular file;

3           object observer program code, in said each of said slave nodes, for observing  
4 change from said particular file version variable to said updated file version variable; and,

5           said commanding program code including program code, responsive to operation  
6 of said object observer program code, for downloading said updated file from said  
7 originator node into said particular file in said each of said slave nodes.

8  
9   33.    The computer program product of claim 27 and wherein said replicating program  
10 code comprises:

11           local workspace program code for receiving said updated file in said originator  
12 node; and,

13           global workspace program code, operatively coupled to said local workspace  
14 program code, for receiving said updated file from said local workspace program code in  
15 preparation to download said updated file to any of said slave nodes upon request from  
16 said any of said slave nodes.

17  
18   34.    The computer program product of claim 33 and wherein said replicating program  
19 code further comprises:

20           at least one data file including a data word with its corresponding file version  
21 variable.

22  
23   35.    The computer program product of claim 27 and further comprising:

1           program code for associating a file version variable with said updated file;  
2           said storing program code comprising:  
3                 local workspace program code for receiving said updated file in said  
4           master node;  
5                 program code for error checking said file version variable to confirm  
6           validity of said file version variable;  
7                 global workspace program code adapted to receive said updated file from  
8           said local workspace program code; and,  
9                 program code, responsive to operation of said error checking program  
10          code confirming said validity, for transferring said updated file to said global  
11          workspace program code; and,  
12          program code, responsive to operation of said transferring program code, for  
13          communicating both creation of said backup file to said originator node and availability  
14          of said backup file to said slave nodes.

15  
16   36.     The computer program product of claim 33 and wherein said local workspace  
17   program code further includes:

18           multiple source program code for receiving said updated file from a new-data-  
19   supplier group consisting of a network user, a security provider, and other providers.

20  
21   37.     The computer program product of claim 36 and wherein said multiple source  
22   receiving program code includes additional program code for receiving additional  
23   updated files from said new-data-supplier group.

1  
2 38. The computer program product of claim 24 and wherein said receiving program  
3 code is further capable of receiving multiple updated files, each of said files being  
4 received from a different network user.

5  
6 39. The computer program product of claim 27 wherein said replicating program code  
7 includes program code for establishing a particular file version variable corresponding to  
8 said particular file, said computer program product further comprising:

9 polling program code for allowing said each of said slave nodes to periodically  
10 poll said master node to determine if said particular file contents matches said updated  
11 file contents; and,

12 synchronizing program code, responsive to operation of said polling program  
13 code determining that said particular file contents do not match said updated file contents,  
14 for synchronizing said particular file contents with said updated file contents.

15  
16 40. The computer program product of claim 39 further comprising:

17 program code for establishing a DPS version number to identify the current  
18 version of DPS in said network;

19 program code for establishing a DDB version number to identify the current  
20 version of DDB in said network;

21 first program code for comparing said DPS version number on said each of said  
22 slave nodes with said DPS version number on said master node to obtain a respective  
23 DPS version number match;

1           said synchronizing program code,  
2                 responsive to operation of said first comparing program code obtaining  
3           said respective DPS version number match on each of certain of said slave nodes  
4           for terminating further operation of said synchronizing program code with respect  
5           to the current said poll on said each of said certain of said slave nodes, and  
6                 responsive to operation of said first comparing program code not obtaining  
7           said respective DPS version number match on each of the remainder of said slave  
8           nodes for achieving said DDB version number match on said each of the  
9           remainder of said slave nodes;  
10          second program code, responsive to operation of said first comparing program  
11   code not obtaining said respective DPS version number match, for comparing said  
12   particular file version variable on said each of the remainder of said slave nodes with said  
13   updated file version variable on said master node to obtain a respective file version  
14   variable match; and,  
15          said synchronizing program code,  
16                 responsive to operation of said second comparing program code obtaining  
17          a file version variable match on each of a portion of said remainder of said slave  
18          nodes for terminating operation of said synchronizing program code with respect  
19          to the current said poll on said each of said portion, and  
20                 responsive to operation of said second program code not obtaining a file  
21          version variable match on each of the remaining portion of said remainder of said  
22          slave nodes for achieving said file version variable match on said each of the  
23          remaining portion;

1           whereby said particular file contents matches said updated file contents in said  
2 each of said slave nodes.  
3

4   41.    The computer program product of claim 27 wherein said replicating program code  
5 includes program code for establishing a particular file version variable corresponding to  
6 said particular file, and wherein each of said plurality of nodes is a storage computer  
7 program product having storage media on which both said particular file version variable  
8 and said particular file contents are stored, said computer program product further  
9 comprising:

10           polling program code, for comparing said particular file version variable stored on  
11 said storage media in said each of said nodes with said particular file version variable  
12 stored elsewhere in its respective node to determine a particular file version variable  
13 match for said each of said nodes; and

14           synchronizing program code, responsive to operation of said polling program  
15 code determining said particular file version variable match was not achieved for certain  
16 of said nodes for achieving said particular file version variable match for each of said  
17 certain nodes.  
18

19   42.    The computer program product of claim 41 and wherein said storage media is at  
20 least one storage disk.  
21

22   43.    The computer program product of claim 29 further comprising:

1           program code, responsive to operation of said commanding program code not  
2   obtaining said updated file from said originator node, for further commanding said  
3   respective node to obtain said backup file from said master node.

4  
5   44.    The computer program product of claim 35 further comprising:

6           program code, responsive to operation of said error checking program code not  
7   confirming said validity, for flagging an error, stopping operation of said file replicating  
8   program code on said updated file, and preparing said file replication computer program  
9   product to receive a next successive updated file.

10  
11  
12   45.    In a computer network having a plurality of nodes, a file replication method  
13   comprising:

14           arranging for receipt of a file in any one of said nodes;  
15           receiving said file in a certain one of said nodes; and,  
16           replicating said file in all other of said nodes.

17  
18   46.    The method of claim 45 wherein said file is an updated file and wherein said  
19   replicating program code includes:

20           replicating said updated file in all other of said nodes in a manner that is network-  
21   topology independent.

22  
23   47.    The method of claim 46 and wherein said replicating program code includes:



1 replicating said updated file in all other of said nodes in a manner that avoids a  
2 single point of failure.

3  
4 48. The method of claim 46 and wherein said certain one of said nodes is the  
5 originator node.

6  
7 49. The method of claim 48 further comprising:  
8 establishing another of said nodes as master node, said plurality of nodes except  
9 for both said originator node and said master node being slave nodes;  
10 storing said updated file on said master node as a backup file; and,  
11 in each of said slave nodes, updating a particular file corresponding to said  
12 updated file.

13  
14 50. The method of claim 49 further comprising:  
15 communicating both creation of said backup file to said originator node and  
16 availability of said backup file to said slave nodes.

17  
18 51. The method of claim 50 further comprising:  
19 publishing a representation of said updated file to said slave nodes; and  
20 within each of said slave nodes, commanding its respective node to obtain said  
21 updated file from said originator node.

52. The method of claim 50 and wherein communicating said creation of said backup  
file to said originator node comprises a success note.

53. The method of claim 51 further comprising:  
establishing an updated file version variable as said representation of said updated  
file; and  
publishing said updated file version variable to said slave nodes.

54. The method of claim 53 further comprising:  
establishing a particular file version variable corresponding to said particular file;  
observing change from said particular file version variable to said updated file  
version variable; and,  
downloading said updated file from said originator node into said particular file in  
said each of said slave nodes.

55. The method of claim 49 further comprising:  
receiving said updated file in local workspace of said originator node; and,  
receiving said updated file into global workspace from said local workspace in  
preparation to download said updated file to any of said slave nodes upon request from  
said any of said slave nodes.

56. The method of claim 55 and wherein said replicating further comprises:

1 including a data word with its corresponding file version variable in at least one  
2 data file.

3  
4 57. The method of claim 49 further comprising:

5 associating a file version variable with said updated file;

6 said storing comprising:

7 receiving said updated file in local workspace of said master node;

8 error checking said file version variable to confirm validity of said file

9 version variable;

10 adapting global workspace to receive said updated file from said local

11 workspace; and,

12 transferring said updated file to said global workspace responsive to said

13 error checking confirming said validity; and,

14 communicating both creation of said backup file to said originator node and

15 availability of said backup file to said slave nodes responsive to said transferring.

16  
17 58. The method of claim 55 further comprising:

18 receiving said updated file from a new-data-supplier group consisting of a

19 network user, a security provider, and other providers.

20  
21 59. The method of claim 58 further comprising:

22 receiving additional updated files from said new-data-supplier group.

1     60.     The method of claim 46 further comprising:

2             receiving multiple updated files, each of said files being received from a different  
3     network user.

5     61.     The method of claim 49 further comprising:

6             establishing a particular file version variable corresponding to said particular file;  
7             said each of said slave nodes periodically polling said master node to determine if  
8     said particular file contents matches said updated file contents;  
9             determining that said particular file contents do not match said updated file  
10    contents; and,  
11             synchronizing said particular file contents with said updated file contents.

13    62.     The method of claim 61 further comprising:

14             establishing a DPS version number to identify the current version of DPS in said  
15    network;  
16             establishing a DDB version number to identify the current version of DDB in said  
17    network;  
18             first comparing said DPS version number on said each of said slave nodes with  
19    said DPS version number on said master node to obtain a respective DPS version number  
20    match;  
21             said synchronizing,  
22             responsive to said first comparing obtaining said respective DPS version  
23    number match on each of certain of said slave nodes, terminating further

1 operation of said synchronizing with respect to the current poll on said each of  
2 said certain of said slave nodes, and  
3 further responsive to said first comparing not obtaining said respective  
4 DPS version number match on each of the remainder of said slave nodes,  
5 achieving said DDB version number match on said each of the remainder of said  
6 slave nodes;  
7 second comparing said particular file version variable on said each of the  
8 remainder of said slave nodes with said updated file version variable on said master node  
9 to obtain a respective file version variable match responsive to said first comparing not  
10 obtaining said respective DPS version number match; and,  
11 said synchronizing,  
12 further responsive to said second comparing obtaining a file version  
13 variable match on each of a portion of said remainder of said slave nodes,  
14 terminating operation of said synchronizing with respect to said current poll on  
15 said each of said portion, and  
16 further responsive to said second comparing not obtaining a file version  
17 variable match on each of the remaining portion of said remainder of said slave  
18 nodes, achieving said file version variable match on said each of the remaining  
19 portion;  
20 whereby said particular file contents matches said updated file contents in said  
21 each of said slave nodes.  
22

1 63. The method of claim 49 wherein each of said plurality of nodes is a storage  
2 system having storage media and wherein said replicating further comprises:  
3 establishing a particular file version variable corresponding to said particular file;  
4 storing said particular file version variable and said particular file contents on said  
5 storage media to obtain stored particular file version variable and contents;  
6 third comparing said particular file version variable stored on said storage media  
7 in said each of said nodes with said particular file version variable stored elsewhere in its  
8 respective node to determine a particular file version variable match for said each of said  
9 nodes; and,  
10 synchronizing, responsive to third comparing determining said particular file  
11 version variable match was not achieved for certain of said nodes, achieving said  
12 particular file version variable match for each of said certain nodes.  
13

14 64. The method of claim 63 and wherein said storage media is at least one storage  
15 disk.  
16

17 65. The method of claim 51 further comprising:  
18 further commanding said respective node to obtain said backup file from said  
19 master node responsive to said commanding not obtaining said updated file from said  
20 originator node.  
21

22 66. The method of claim 57 responsive to said error checking not confirming said  
23 validity, said method further comprising:

1 flagging an error;

2 stopping said file replicating on said updated file; and,

3 preparing said file replication method to receive a next successive updated file.

4  
5  
6 67. File replication apparatus for use in a computer network having a plurality of  
7 nodes, said apparatus comprising:

8 first file apparatus, distributed on each of said nodes, configured to receive an  
9 updated file in any one of said nodes; and,

10 second file apparatus, distributed throughout said plurality of nodes and  
11 responsive to receiving said updated file in a certain one of said nodes, that replicates said  
12 updated file in all other of said nodes in a manner that is network-topology independent  
13 and avoids a single point of failure.  
14  
15

16 68. A method for synchronizing data in a network having a master node and at least  
17 one slave node comprising:

18 establishing a DPS in said master node and said at least one slave node;

19 for each said slave node, periodically polling said master node to determine if the  
20 version number of said DPS in said master node matches the version number of said DPS  
21 in said at least one slave node;

22 for each said slave node, terminating said method if said master node DPS version  
23 number matches said at least one slave node DPS version number; and,

1 if said master node DPS version number does not match said at least one slave  
2 node DPS version number thereby providing a mismatch:

3 first determining if said mismatch is due to a DDB version number  
4 mismatch only and, if so, first replicating IP addresses of said at least one slave node and  
5 terminating said method; and,

6 if said mismatch is not due to a DDB version number mismatch only,  
7 second determining if said mismatch is due to a FVV mismatch only and, if so, second  
8 replicating new files associated with a new said FVV and terminating said method; and,

9 if said mismatch is due to both said DDB version number mismatch and  
10 said FVV mismatch, performing said first replicating and said replicating and terminating  
11 said method.

12  
13  
14 69. A method for synchronizing data on storage media in a network having a master  
15 node and at least one other node including slave node and originator node, said method  
16 comprising:

17 establishing a DPS in said master node and said at least one other node;

18 for each node in said network, periodically comparing each FVV stored in its  
19 respective said storage media with its corresponding FVV stored in its DPS;

20 if there is a match between said storage media FVV and its corresponding said  
21 DPS FVV, terminating said method;



1 if there is no match between said storage media FVV and its corresponding said  
2 DPS FVV thereby providing a first mismatch, first determining if said first mismatch  
3 occurred on said master node or on said at least one other node;

4 if said first mismatch occurred on said at least one other node second determining  
5 if said first mismatch is due to a missing file or an extra file on said other node storage  
6 media; and

7 if due to said missing file, retrieving said missing file from said originator  
8 node or said master node and terminating said method; and,

9 if due to said extra file, ignoring said extra file and deleting said extra file  
10 from said media and terminating said method; and,

11 if said first mismatch occurred on said master node, third determining if said  
12 mismatch is due to a missing file or an extra file on said master node storage media; and

13 if due to said missing file, removing the corresponding file from said DPS  
14 in said master node and terminating said method; and,

15 if due to said extra file, adding said extra file to said DPS in said master  
16 node and terminating said method.

17  
18  
19 70. In a computer network having a plurality of nodes, a file replication system  
20 comprising:

21 means, distributed on each of said nodes, capable of receiving a new file in any  
22 one of said nodes; and,

1 means, distributed throughout said plurality of nodes and responsive to receiving  
2 said new file in a certain one of said nodes, for replicating said new file in all other of  
3 said nodes in a manner that is network topology independent and avoids a single point of  
4 failure.

5  
6  
7 71. A computer program product for use in a computer network having a plurality of  
8 nodes, said computer program product including a computer usable medium having  
9 computer readable program code thereon for file replication, said program code  
10 comprising:

11 program code, distributed on each of said nodes, capable of receiving a new  
12 file in any one of said nodes; and,

13 program code, distributed throughout said plurality of nodes and responsive to  
14 receiving said new file in a certain one of said nodes, for replicating said new file in all  
15 other of said nodes in a manner that is network topology independent and avoids a single  
16 point of failure.

17  
18  
19 72. In a computer network having a plurality of nodes, a file replication method  
20 comprising:

21 arranging for receipt of a new file in any one of said nodes;

22 receiving said new file in a certain one of said nodes; and,

8